You will need some conditions to claim the equivalence between minimizing cross entropy and minimizing KL divergence. I will put your question under the context of classification problems using cross entropy as loss functions.

Let us first recall that entropy is used to measure the uncertainty of a system, which is defined as

$$S(v)=-\sum_i p(v_i)\log p(v_i),$$

for $p(v_i)$ as the probabilities of different states v_i of the system. From an information theory point of view, $S(v)$ is the amount of information is needed for removing the uncertainty.

For instance, the event I I will die within 200 years is almost certain (we may solve the aging problem for the word *almost*), therefore it has low uncertainty which requires only the information of the aging problem cannot be solved to make it certain. However, the event II I will die within 50 years is more uncertain than event I, thus it needs more information to remove the uncertainties. Here entropy can be used to quantify the uncertainty of the distribution When will I die?, which can be regarded as the expectation of uncertainties of individual events like I and II.

Now look at the definition of KL divergence between distributions A and B

$$D_{KL}(A\|B)=\sum_i p_A(v_i)\log p_A(v_i)-p_A(v_i)\log p_B(v_i),$$

where the first term of the right hand side is the entropy of distribution A, the second term can be interpreted as the expectation of distribution B in terms of A. And the D_{KL} describes how different B is from A from the perspective of A. It's worth of noting A usually stands for the data, i.e. the measured distribution, and B is the theoretical or hypothetical distribution. That means, you always start from what you observed.

To relate cross entropy to entropy and KL divergence, we formalize the cross entropy in terms of distributions A and B as

$$H(A,B)=-\sum_i p_A(v_i)\log p_B(v_i).$$

From the definitions, we can easily see

$$H(A,B)=D_{KL}(A\|B)+S_A.$$

If S_A is a constant, then minimizing $H(A,B)$ is equivalent to minimizing $D_{KL}(A\|B)$.

A further question follows naturally as how the entropy can be a constant. In a machine learning task, we start with a dataset (denoted as $P(D)$) which represent the problem to be solved, and the learning purpose is to make the model estimated distribution (denoted as $P(model)$) as close as possible to true distribution of the problem (denoted as $P(truth)$). $P(truth)$ is unknown and represented by $P(D)$. Therefore in an ideal world, we expect

$$P(model)\approx P(D)\approx P(truth)$$

and minimize $D_{KL}(P(D)\|P(model))$. And luckily, in practice D is given, which means its entropy $S(D)$ is fixed as a constant.